LIP

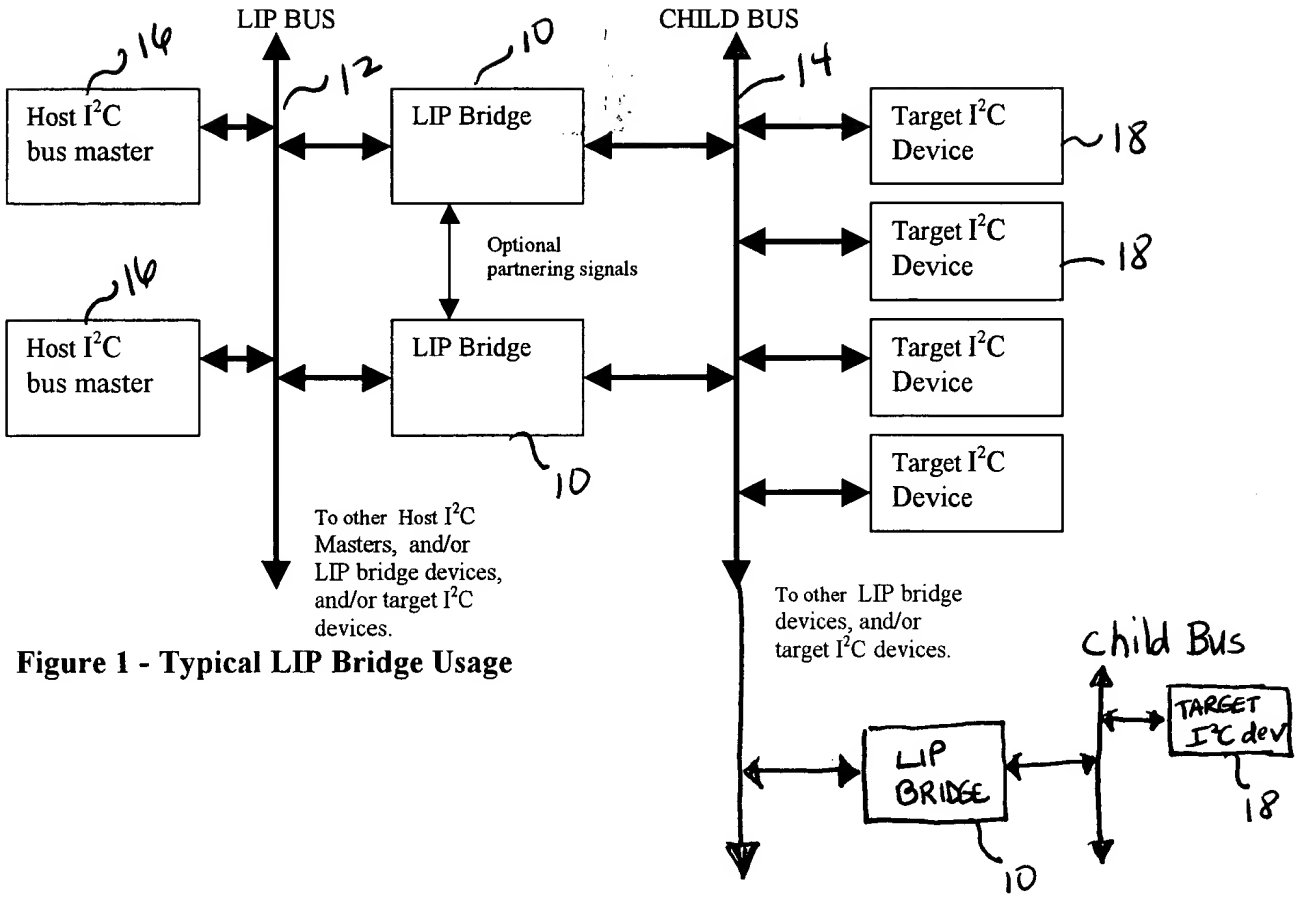


Figure 1 - Typical LIP Bridge Usage

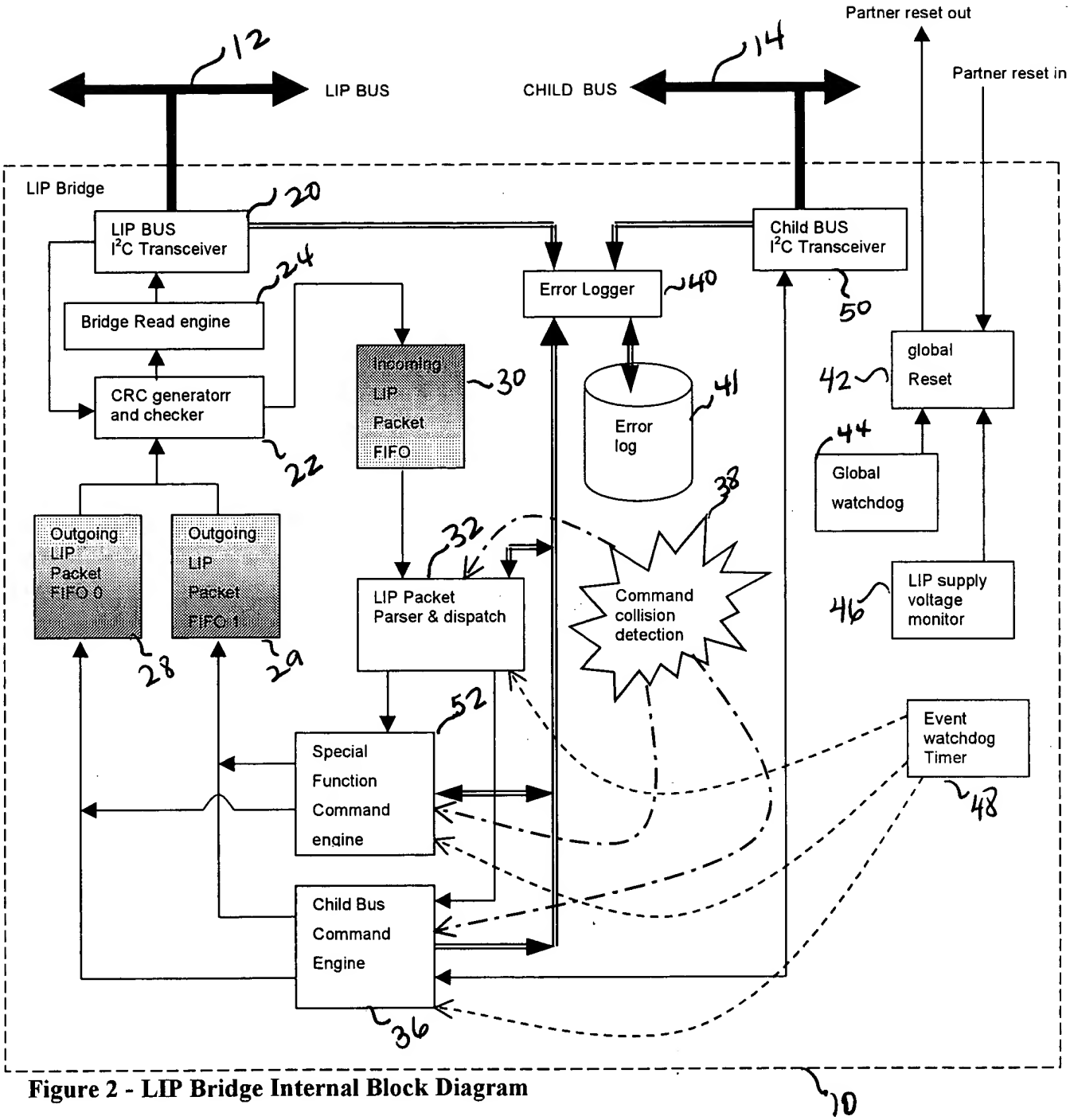
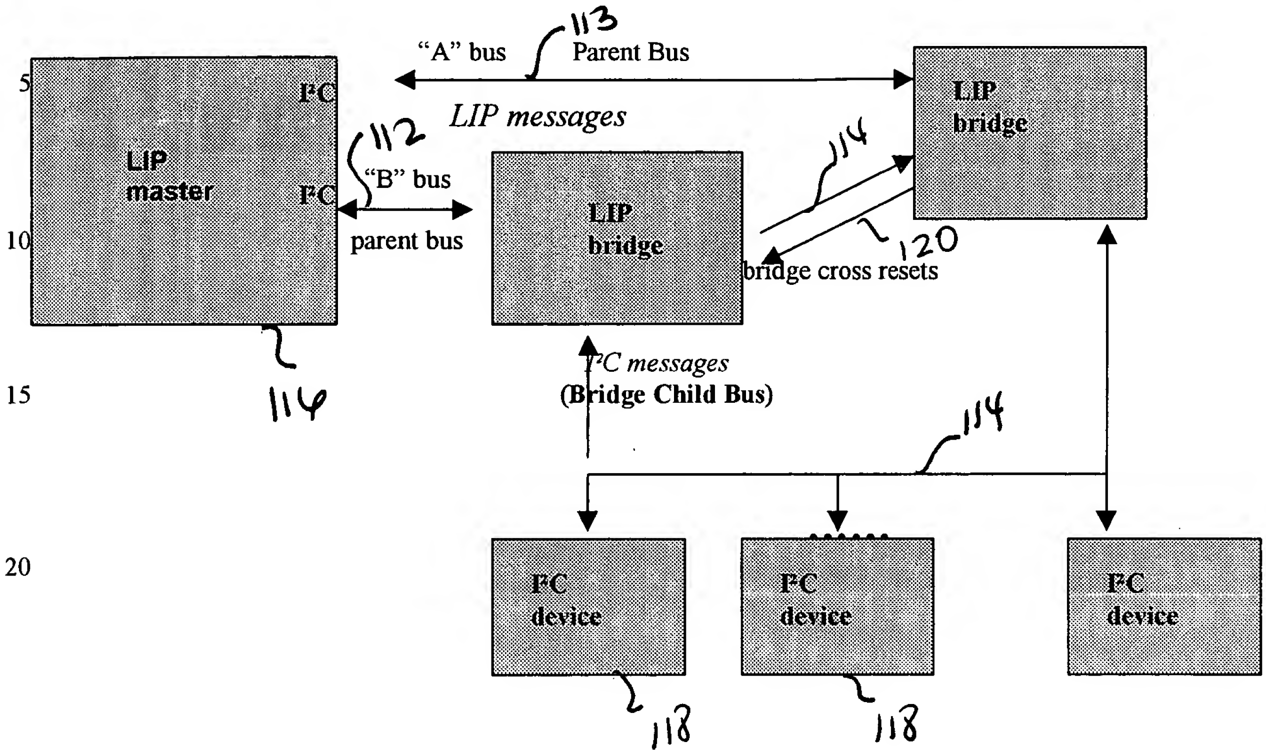


Figure 2 - LIP Bridge Internal Block Diagram

Figure 3: Typical Usage of LIP Bridge



4/13

LIP address + W	Child Address +R/W or Function	Data for Write -or- Read Count	CRC

122 124 126 128

FIG. 4

130

7	6	5	4	3	2	1	0
A6 MSB	**A5**	**A4**	**A3**	**A2**	**A1**	**A0 LSB**	**Odd parity**

FIG. 5 Hardware Address Strapping

The LIP Address / Function encoding within the four byte LIP packet is as follows:

LIP address

7	6	5	4	3	2	1	0
MSB. A6	**A5**	**A4**	**A3**	**A2**	**A1**	**A0 LSB**	**R/!W**

LIP Address	Child address / Function	RdCnt / Fnc code	CRC

122 124 126 128

FIG. 6

The Child Address / Function encoding is as follows:

Child Address / Function

7	6	5	4	3	2	1	0
MSB	...					**LSB**	**R/!W**

LIP Address	*Child address / Function*	Wr data / RdCnt / Fnc code	CRC

122 124 126 128

FIG. 7

Read Count Field

7 MSB	6	5	4	3	2	1	0 LSB
Rsvd - 0	**SrcId**	**RdCnt**					

LIP Address	Child Bus address	*Read Count field*	CRC

122 124 126 128

FIG. 8

Read Data Tag

7 MSB	6	5	4	3	2	1	0 LSB
NoData	**SrcId**	**RdCnt**					

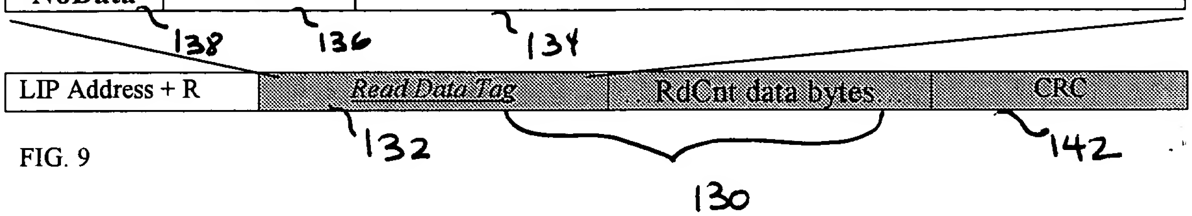


FIG. 9

Status Byte Register Organization

RAZ	***RAZ***	***RAZ***	**ME**	**LBWE**	**LBRE**	**CBWE**	**CBRE**
MSB							*LSB*

150

FIG. 10

Fig. 10

Table 8

Key to Symbols

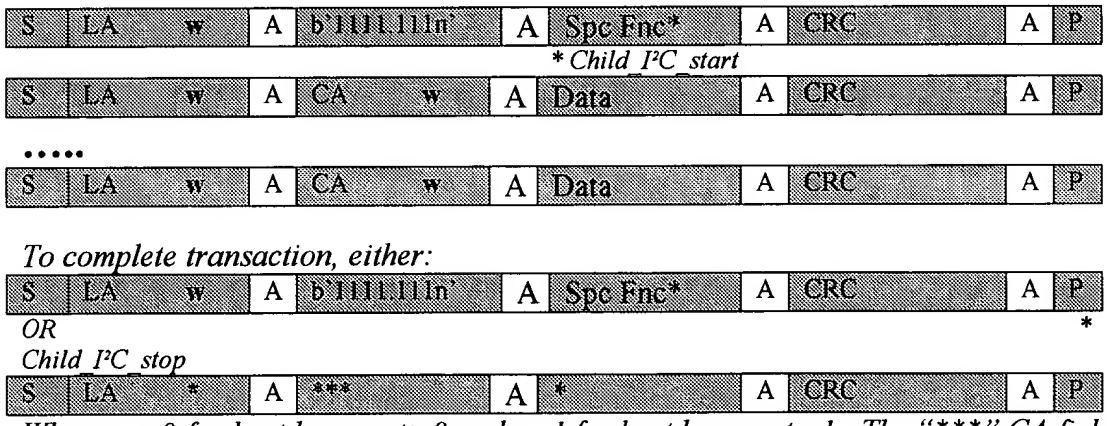
Symbol	Meaning
S	I²C bus start condition
P	I²C bus stop condition
A	Acknowledge
A	No-Acknowledge
LA	LIP address
CA	Child bus address
W	R/!W bit within address field is set for WRITE
R	R/!W bit within address field is set for READ
CRC	CRC byte
Data	Data byte
Count	Read count
Fnc(x)	Special function command "x" – where x is the function's hex code
	Gray shade indicates data sent from host bus master to LIP bridge
	White indicates data sent from LIP bridge to host bus master
.....	Zero or more instances of the preceding transaction.

Host bus master to LIP One Byte Child Bus Write

160

FIG. 11

Host bus master to LIP Multi-Byte Write



162

*Where: n=0 for host bus master0 and n=1 for host bus master1. The "***" CA field contains a different value than that contained in the first data packet (SECOND PACKET ABOVE). This can be either a different child address, or it can be a special function indicator of binary '1111.111n'.*

FIG. 12

Special Function Action Neither Requiring Nor Returning Data

S	LA w	A	b'1111.111n'	A	Fnc(x)	A	CRC	A	P

Where n=0 for host bus master0, n=1 for host bus master1 & "x" is the desired hex command code.

164

FIG. 13

Special Function Action Returning Data

S	LA w	A	b'1111.111n'	A	Fnc(x)	A	CRC	A	P

Where n=0 for host bus master0, n=1 for host bus master1 & "x" is the desired hex command code.

S	LA r	A	Read data tag	A	Data	A	....	Data	A	CRC	a	P

166

FIG. 14

Special Function Action Requiring Data

S	LA w	A	0xFE	A	Data	A	CRC	A	P

168

FIG. 15

Host bus master Child Bus Read Via LIP Bridge Action

S	LA w	A	CA r	A	Read Count	A	CRC	A	P

S	LA r	A	Read data tag	A	Data	A	....	Data	A	CRC	a	P

170

FIG. 16

PLCC

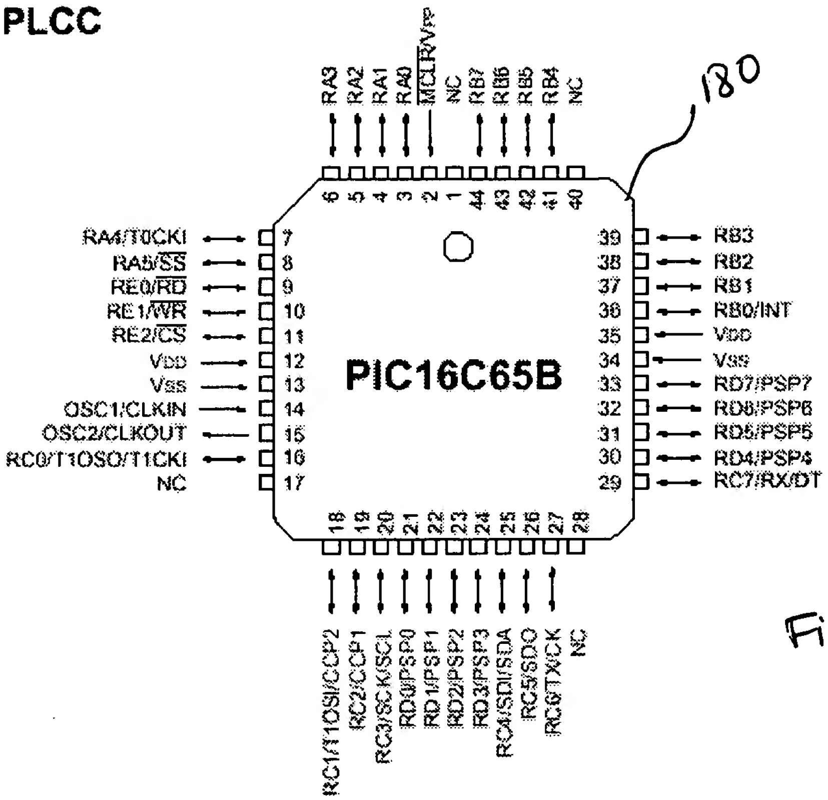


Fig. 17

PIN #	Label	Function
2	!MCLR	!partner_reset_in -- Active low input for reset from partner LIP bridge (Also VPP pin for in circuit programming)
3	RA0	!partner_reset_out – Active low output to reset partner LIP bridge
20	RC3	LIP_clk -- LIP bus serial clock in
25	RC4	LIP_data -- LIP bus serial data in/out (bidirectional)
27	RC6	Child_clk -- child bus clock output
29	RC7	Child_data – child bus data in/out (bidirectional)
37	RB1	LIP_addr_parity – parity bit for LIP address (strap to make odd parity)
38	RB2	LIP_addr0 – bit 0 to strap LIP I²C address
39	RB3	LIP_addr1 – bit 1 to strap LIP I²C address
41	RB4	LIP_addr2 – bit 2 to strap LIP I²C address
42	RB5	LIP_addr3 – bit 3 to strap LIP I²C address
4	RA1	LIP_addr4 – bit 4 to strap LIP I²C address
5	RA2	LIP_addr5 – bit 5 to strap LIP I²C address
43	RB6	In circuit programming clock
44	RB7	In circuit programming data
6	RA3	Child_bus_busy_out –active low open collector output when this LIP bridge owns child bus (needs a 1K pull up to Vdd).
36	RB0	Child_bus_busy_in –active low input when partner LIP bridge owns child bus

9/13

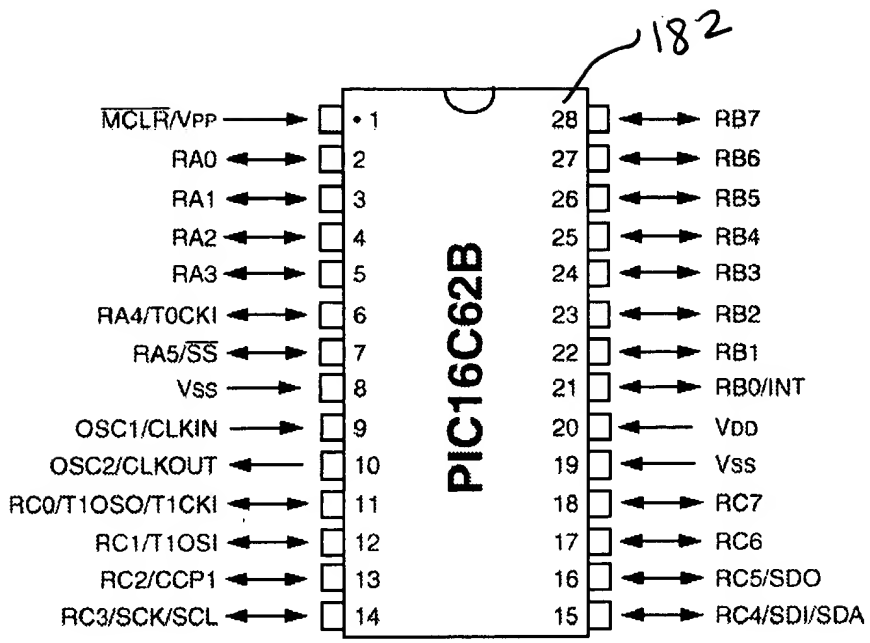


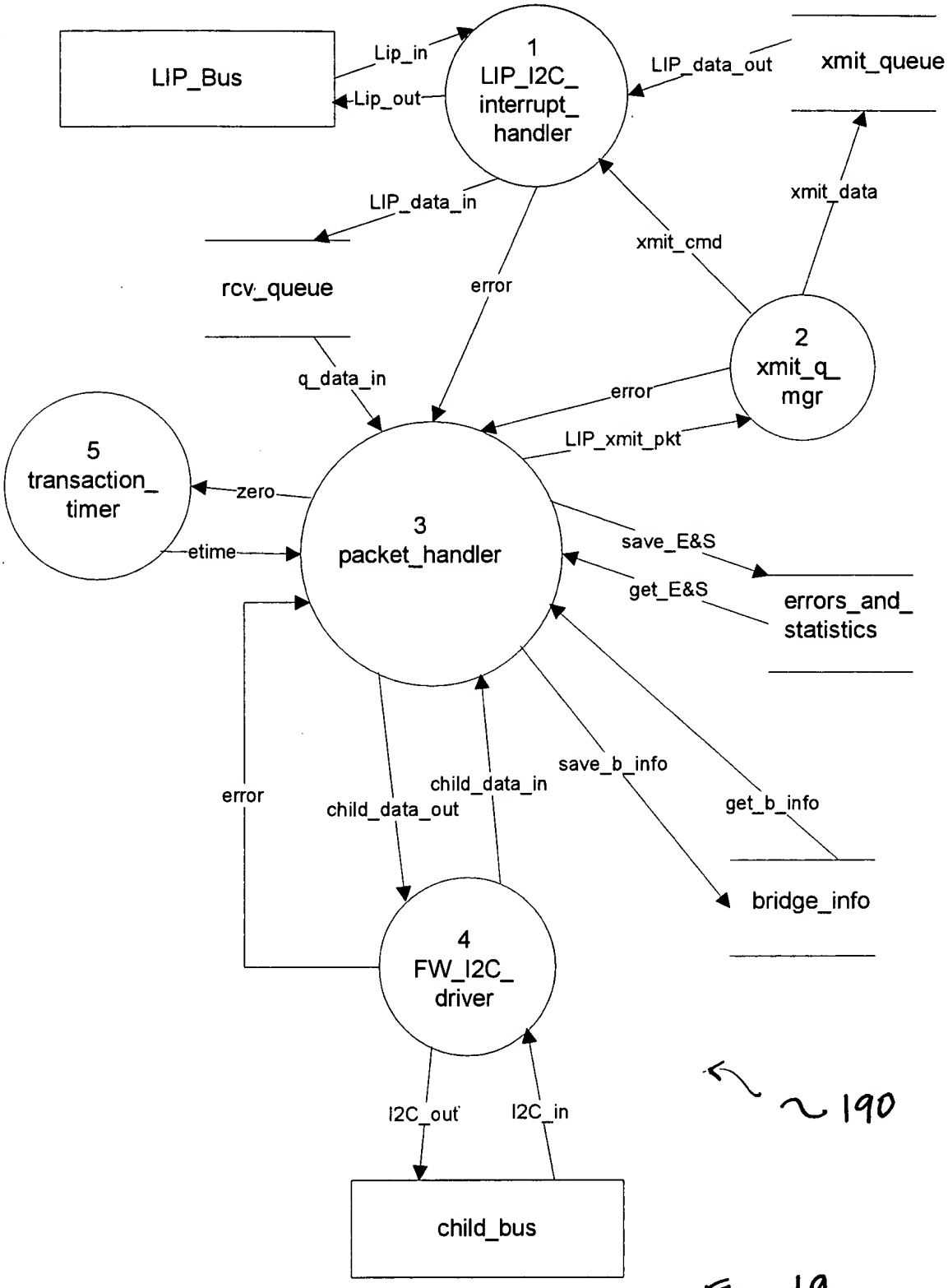
Fig 18

PIN #	Label	Function
1	!MCLR	!partner_reset_in -- Active low input for reset from partner LIP bridge (Also VPP pin for in circuit programming)
2	RA0	!partner_reset_out – Active low output to reset partner LIP bridge
14	RC3	LIP_clk -- LIP bus serial clock in
15	RC4	LIP_data -- LIP bus serial data in/out (bidirectional)
17	RC6	child_clk -- child bus clock output
18	RC7	child_data – child bus data in/out (bidirectional)
22	RB1	LIP_addr_parity – parity bit for LIP address (strap to make odd parity)
23	RB2	LIP_addr0 – bit 0 to strap LIP I²C address
24	RB3	LIP_addr1 – bit 1 to strap LIP I²C address
25	RB4	LIP_addr2 – bit 2 to strap LIP I²C address
26	RB5	LIP_addr3 – bit 3 to strap LIP I²C address
3	RA1	LIP_addr4 – bit 4 to strap LIP I²C address
4	RA2	LIP_addr5 – bit 5 to strap LIP I²C address
27	RB6	In circuit programming clock
28	RB7	In circuit programming data
5	RA3	child_bus_busy_out –active low output when this LIP bridge owns child bus (needs a 1K pull up to Vdd).
21	RB0	child_bus_busy_in –active low input when partner LIP bridge owns child bus

5

10/13

Level 1 Data Flow Diagram



~190

Fig. 19

Packet Parser Data Flows

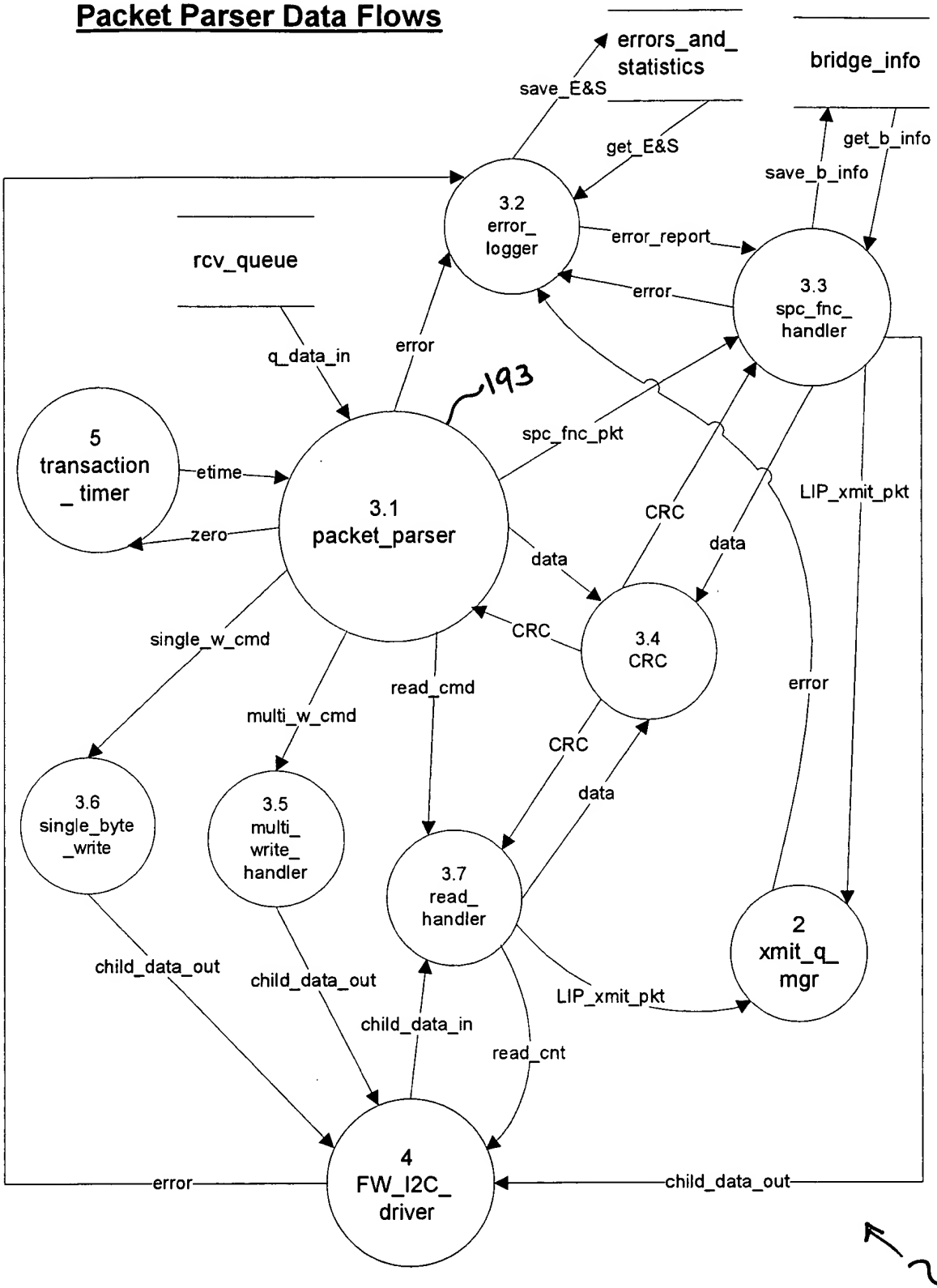


Fig. 20

Firmware I2C Data Flows

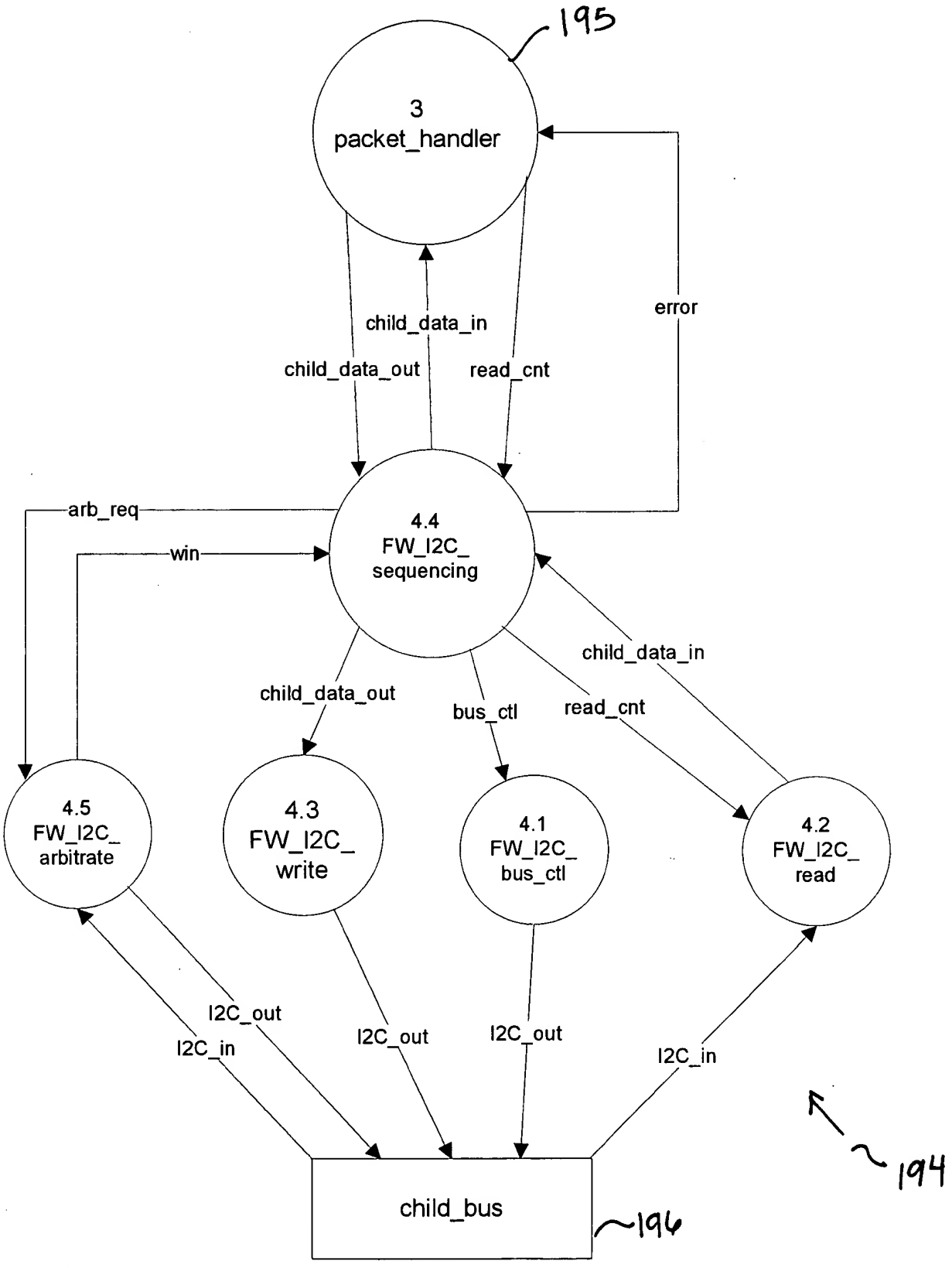


Fig. 21

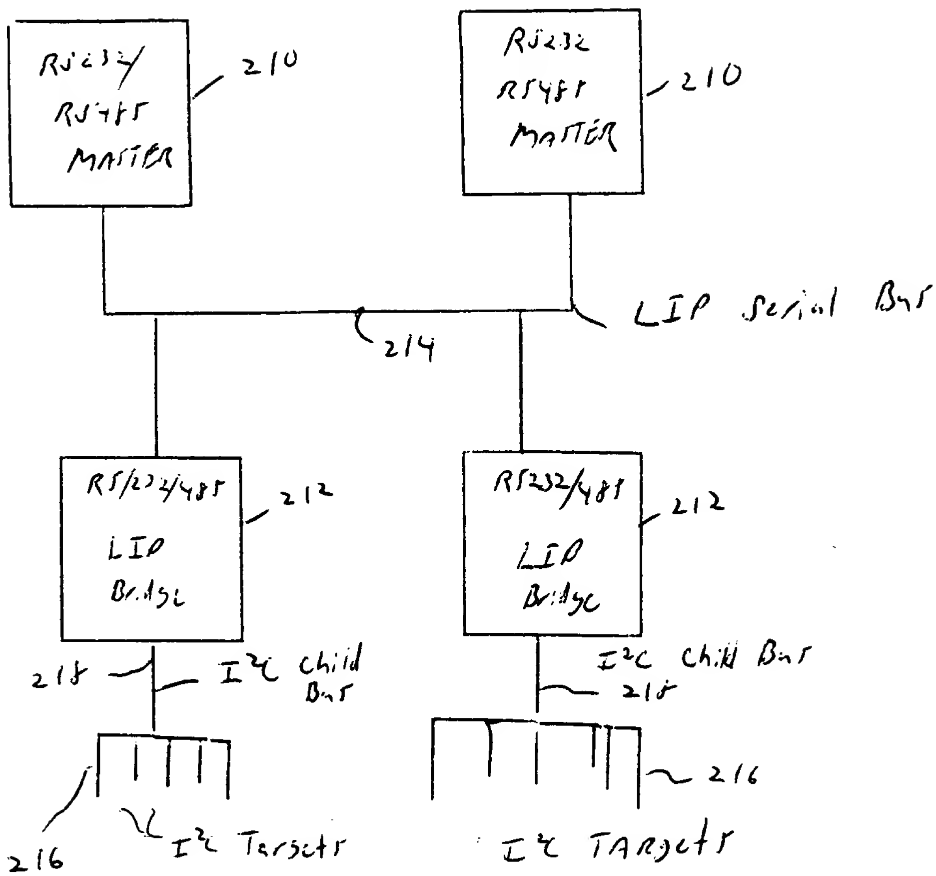

RS232/
RS485
MASTER
210
RS232
RS485
MASTER
210
LIP Serial Bus
214
RS/232/485
LIP
Bridge
212
RS232/485
LIP
Bridge
212
218
I²C child Bus
I²C Child Bus
218
216
216
I²C Targets
I²C Targets

Fig 22